Doc Code: AP.PRE.REQ

PTo/SB/33 (07-05)
Approved for use through xx/xx/200x. OMB 0651-00xx
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

PRE-APPEAL BRIEF REQUEST FOR REVIEW

Docket Number (Optional)	
JRL-550-510	

Application Number	Filed
10/764,495	January 27, 2004
First Named Inventor: NIGHTINGALE	
Art Unit: 2128	Examiner: Alhija, Saif A.

Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request.

This request is being filed with a notice of appeal.

The review is requested for the reason(s) stated on the attached sheet(s).
Note: No more than five (5) pages may be provided.

I am the

☐ Applicant/Inventor

☐ Assignee of record of the entire interest. See 37 C.F.R. § 3.71. Statement under 37 C.F.R. § 3.73(b) is enclosed. (Form PTO/SB/96)

☒ Attorney or agent of record 33,149 (Reg. No.)

☐ Attorney or agent acting under 37CFR 1.34.
Registration number if acting under 37 C.F.R. § 1,34 _____


Signature

John R. Lastova
Typed or printed name

703-816-4025
Requester's telephone number

March 30, 2007
Date

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below.*

☒ *Total of 1 form/s are submitted.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of

NIGHTINGALE

Atty. Ref.: 550-510; Confirmation No. 2384

Appl. No. 10/764,495

TC/A.U. 2128

Filed: January 27, 2004

Examiner: Alhija, Saif A.

For: APPARATUS AND METHOD FOR PERFORMING HARDWARE AND SOFTWARE CO-VERIFICATION TESTING

* * * * * * * * * * * *

March 30, 2007

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

PRE-APPEAL BRIEF REQUEST FOR REVIEW

Clear Error #1: The Claims Define Statutory Subject Matter

The claims define statutory subject matter. For example, claim 1 recites an "apparatus for performing a sequence of verification tests to perform hardware and software co-verification on a system under verification." The "apparatus" statutory category is well recognized, and the recited apparatus elements include signal interface controllers coupled to the system under verification, a processing unit that executes software routines, a debugger that controls the processing unit, a debugger signal interface controller that causes stimulus and response signals to be exchanged between the debugger and the debugger signal interface controller, and a test manager that accomplish the hardware and software co-verification on a system under verification. How can the physical structure recited in claim 1 that among other things exchanges physical stimulus and response signals between two of the elements be "software per se" as alleged by the Examiner?

Method claim 18 describes steps of physically testing a system using "signal interface controllers" and "controlling via a debugger execution of software routines by a processing unit associated with the system under verification." Physical stimulus signals and response signals are communicated between the controllers and the system under test. How can the physical operations recited in claim 18 performed by a physical processing unit using physical signals on a physical system under verification be "software per se" as alleged by the Examiner?

Claim 35 defines "a computer program product embodied on a computer-readable medium for use in performing a sequence of verification tests to perform hardware and software co-verification on a system under verification." Each program code element specifically includes language explaining that the "code, when executed, causes the computer to" perform some specific operation. How can the claimed computer program code embodied on a computer-readable medium, (a physical structure), which when executed causes a computer to perform the recited tasks be "software per se" as alleged by the Examiner?

The pre-appeal board is requested to carefully consider the arguments and case law analysis set forth in the October 18, 2006 amendment on pages 12-14. The claimed apparatus, method, and software product solve a very real technical problem and significantly improve the efficiency of designing a data processing system such as a System-on-Chip (SoC). Moreover, at least one or more of the signal interface controller, debugger, debugger signal interface controller, and test manager in claim 1, as well as the at least one or more of the steps in method claim 18, is implemented in a suitably programmed computer, which is tangible hardware. The §101 rejection is improper and should be withdrawn.

Clear Error #2: <u>Rajsuman Lacks A Debugger For Driving A Processing Unit In A System Under Verification In Order To Co-Ordinate The Execution Of Software Routines With The Sequence Of Verification Tests</u>

Claim 1 recites "a debugger operable to control operation of a processing unit associated with the system under verification, the processing unit being operable to execute software routines." When performing a sequence of verification tests with respect to the system under verification, the test manager issues test control messages to signal interface controllers coupled to that system and to a debugger signal interface controller. The test controlling messages cause the debugger signal interface controller to perform one or more test actions during which stimulus signals and response signals are passed between the debugger and the debugger signal interface controller during performance of the sequence of verification tests. This allows the test manager to *coordinate* the execution of the *software* routines by the processing unit *along with* the *sequence of verification tests*. All the features of claim 1 are not taught by Rajsuman.

In referring to col. 5, lines 43 to 44, the Examiner apparently equates the verification units of Figure 5 in Rajsuman with the claimed signal interface controllers. Figure 5 shows that each verification unit is associated with a separate design validation station (DVS) which together represent a SoC having a plurality of functional cores. Col.1, lines 6 to 8. The Examiner refers to column 11, lines 53 to 63 for the claimed debugger. This text describes timing verification of critical paths of the SoC design. If the test results obtained using the vectors in the testbench data deviate from the simulation timing results, then there is an error that needs debugging. So this text in Rajsuman simply describes standard debugging activity. There is no teaching in Rajsuman of using a debugger to *control operation* of a processing unit

included in the hardware being tested which is also executing software routines as part of a software verification test.

On page 2 of the final action, the Examiner also refers to claim 5 and col. 9, lines 16-24 in Rajsuman with regard to the claimed debugger. Both explain that Rajsuman's system can produce test patterns which can be used for "debugging of a fault in the cores of the SoC." Claim 5. But the claimed debugger functionality is different from some general sort of traditional debugging activity. Rather, the claimed debugger controls the processing unit executing one or more software routines to facilitate hardware and software co-verification, which has traditionally proved to be difficult, as explained in Applicant's prior response and in the application background. The Examiner never identifies where Rajsuman's debugging does this. The claimed debugger is used as a tool to synchronize the hardware and software activities to achieve hardware and software co-verification of the system under test.

Clear Error #3: <u>Rajsuman Lacks The Claimed Debugger Signal Interface Controller</u>

The Examiner refers to the same sections of Rajsuman for the claimed "debugger signal interface controller operable to interface with the debugger and to perform one or more test actions transferring at least one of one or more stimulus signals and one or more response signals between the debugger and said debugger signal interface controller during performance of said sequence of verification tests." But they do not disclose the claimed debugger signal interface controller. The debugger signal interface controller receives test controlling messages from the test manager which identify the test actions to be performed. In response, the debugger signal interface controller performs those test actions to cause stimulus signals and response signals to pass between the debugger signal interface controller and the debugger. Those stimulus and response signals affect what software routine is run by the processing unit because the debugger

controls operation of the processing unit. In this way, the test manager can "control the operation of the processing unit via the debugger signal interface controller and the debugger in order to co-ordinate the execution of said software routines with the sequence of verification tests." None of this is taught by Rajsuman.

As explained above, the general reference in Rajsuman to debugging does disclose the recited functionality in the independent claims. That claimed functionality is not directed to detecting faults in the claimed processing unit, but instead to synchronizing hardware and software activities, thereby enabling a sequence of verification tests to be performed to test correct operation of the software and hardware of the system in combination. Rajsuman lacks the coordinated execution of the claimed software routines along with the sequence of verification tests made possible with the claimed debugger and debugger interface controller.

Given the clear errors set forth above, the final rejection should be withdrawn, and the application passed to allowance.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: _____
John R. Lastova
Reg. No. 33,149

JRL:maa
901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100